

UNIVERSITETET I OSLO

Det matematisk-naturvitenskapelige fakultet

Eksamen i IN 211 — Programmeringsspråk

Eksamensdag: 6. desember 2001

Tid for eksamen: 9.00 – 15.00

Oppgavesettet er på 9 sider.

Vedlegg: Ingen

Tillatte hjelpemidler: Alle trykte og skrevne

Kontroller at oppgavesettet er komplett før du begynner å besvare spørsmålene.

Innhold

1 Kjøresystemer (vekt 25%)	side 2
2 Prolog-grammatikk (vekt 20%)	side 4
3 Regulære uttrykk (vekt 15%)	side 5
4 Bussruter i ML (vekt 30%)	side 6
5 Bussruter i Prolog (vekt 10%)	side 8

Først noen generelle råd og bemerkninger:

- Oppgavesettet består av fem uavhengige deler. Prosentene angitt på hver oppgave antyder hvor mye vekt de forskjellige delene vil bli tillagt ved sensuren. De enkelte deloppgavene kan bli gitt ulik vekt avhengig av vanskelighetsgrad.
- Om oppgaveteksten på noe punkt er uklar eller upresis, kan du gjøre egne presiseringer. Formulér i så fall disse tydelig i oppgavebesvarelsen din.
- Legg vekt på å gi korte og klare forklaringer.
- Les oppgavene nøye!

Lykke til!

Ragnhild Kobro Runde

(Fortsettes på side 2.)

Figur 1: Eksempel-program

```
void main() {
    int a;

    void R() {
        void S() {
            int b;

            void P(int x, int y, int z) {
                y = y + 1;
                z = z + x;
            }

            a = 2; b = 3;
            P(a + b, a, a);
            print(a);
        }
        S();
    }

    void Q() {
        int a;
        void T() {
            a = 5;
        }
        T();
        R();
    }

    a = 1;
    Q();
}
```

Oppgave 1 Kjøresystemer (vekt 25%)

I figur 1 er det vist et lite program skrevet i et C4"-lignende språk (utvidet med parametre). Vi antar at språket har statisk binding av deklarasjoner.

1a Statisk og dynamisk link

Tegn aktiveringsstakken med statisk og dynamisk link, samt variable, slik den ser ut rett *før* kallet på P.

(Fortsettes på side 3.)

1b Parameteroverføring

De fire vanligste mekanismene for parameteroverføring av variable er:

- Verdioverføring
- Referanseoverføring
- Verdi-resultat overføring
- Name-overføring

Se igjen på programmet i figur 1 på forrige side.

1. Hva vil bli skrevet ut av print-setningen ved henholdsvis verdioverføring og name-overføring? Forklar kort hvordan du kom frem til disse verdiene.
2. Anta nå at kallet på P erstattes av kallet $P(b, a, a)$. Angi for både referanseoverføring og verdi-resultat overføring:
 - (a) Hva som blir skrevet ut av print-setningen.
 - (b) Hvordan denne verdien er beregnet.
 - (c) Hvordan aktiveringsstakken ser ut rett *etter* at P er kalt. (Det er tilstrekkelig å angi tillegg/endringer i forhold til stakken i 1a.)
3. Hvorfor er det problematisk å tillate kallet $P(a + b, a, a)$ ved referanseoverføring og verdi-resultat overføring?

1c Objekter

I figur 2 på neste side finner du et lite Java-program. Du skal nå tegne datalageret slik det vil se ut (med statisk link, dynamisk link, variable og objekter) rett før kallet på T returnerer.

Bruk det du har lært i forbindelse med blokkstrukturerte språk til å finne ut hvordan det må bli med statisk og dynamisk link i dette tilfellet. Husk at objekter ikke kan legges på stakken.

(Hint: Tegn stakken med en aktiveringsblokk for `Program` øverst.)

(Fortsettes på side 4.)

Figur 2: Java-program

```

class Program {
    static int a;
    public static void main(String[] args) {
        int b = 1;
        a = 2;
        Q qpeker = new Q();
    }
}
class Q {
    int a;
    void T() {
        a = 5;
    }
    // Konstruktør (kalles når et Q-objekt opprettes):
    Q() {
        T();
    }
}

```

Oppgave 2 Prolog-grammatikk (vekt 20%)

Gitt følgende grammatikk (i utvidet BNF) for en forenklet versjon av Prolog (der $\langle statmt \rangle$ er startsymbolet)¹:

$$\begin{aligned}
 \langle statmt \rangle &\rightarrow \langle fact \rangle \mid \langle rule \rangle \\
 \langle fact \rangle &\rightarrow \langle pred \rangle . \\
 \langle pred \rangle &\rightarrow \mathbf{name} \llbracket (\langle arglist \rangle) \rrbracket ? \\
 \langle rule \rangle &\rightarrow \langle pred \rangle ? \mathbf{: - body} . \\
 \langle arglist \rangle &\rightarrow \mathbf{arg} \mid \mathbf{arg} , \langle arglist \rangle \mid \langle list \rangle \\
 \langle list \rangle &\rightarrow [\langle listelems \rangle] \\
 \langle listelems \rangle &\rightarrow \varepsilon \mid \mathbf{arg} \mid \mathbf{arg} \mathbf{I} \langle list \rangle \mid \mathbf{arg} \mathbf{I} \mathbf{var}
 \end{aligned}$$

name og **var** er leksikalske kategorier, og kan betraktes som grunnsymboler. I denne oppgaven kan også **body** og **arg** behandles som grunnsymboler, mens **I** blir her brukt som skilletegn i lister.

2a Syntaksmengder

Beregn meta-til-tom mengden, samt startmengden og etterfølgermengden til hvert metasymbol.

¹Dette er *ikke* identisk med den delmengden av Prolog som er pensum.

(Fortsettes på side 5.)

2b Ikke LL(1)

Vis at grammatikken ikke er LL(1).

2c LL(1)

Vri på grammatikken slik at den blir LL(1) og vis at den nye grammatikken virkelig er LL(1) ved å beregne de utvidede startmengdene.

2d Recursive descent

Bruk grammatikken du kom frem til i 2c til å lage de “recursive decent”-metodene som trengs for å kunne parsere en (del-)setning generert ut fra metasymbolet $\langle list \rangle$.

Det er ikke nødvendig å implementere eventuelle hjelpemetoder (som for eksempel `lesSymbol`), men forklar kort hvordan disse er tenkt å fungere.

Oppgave 3 Regulære uttrykk (vekt 15%)

Vi skal i denne oppgaven se på hvordan vi kan beskrive beløp i hele norske kroner. Et korrekt skrevet beløp er en sekvens av siffer slik at:

- Den eneste sekvensen som kan starte med 0 er sekvensen som består av dette ene sifferet.
- For å lette lesbarheten *kan* tre og tre sifre grupperes (bakfra) for hvert tusen. (Et beløp på en million kroner kan altså skrives både som 1000000 og som 1.000.000)

3a Regulært uttrykk

Skriv et regulært uttrykk for slike beløp. (For å lette skrivearbeidet kan du la \top stå for et vilkårlig siffer fra 1 til 9.)

3b Ikke-deterministisk automat

Tegn en ikke-deterministisk automat ut fra det regulære uttrykket i 3a.

3c Deterministisk automat

Gjør automaten fra 3b om til en deterministisk automat.

(Fortsettes på side 6.)

Oppgave 4 Bussruter i ML (vekt 30%)

Vi skal i denne oppgaven la en bussrute bli representert som et par (rutenummer, stoppesteder) der rutenummer er et vanlig tall og stoppesteder er en liste med strenger.

I ML kan vi bruke typen

```
type bussrute = int * string list;
```

til dette.

Eksempler på slike bussruter kan da være:

```
val rute1 = (23, ["Lysaker", "Ullevål", "Nygård",  
                "Storo", "Teisen", "Ryen"]);  
val rute2 = (21, ["Aker Brygge", "Carl Berner",  
                "Helsfyr", "Teisen", "Bøler"]);
```

Vi lar ML-typen

```
type rutehefte = bussrute list;
```

representere et rutehefte som en liste med bussruter. Et eksempel på et slikt rutehefte kan da være:

```
val ruter = [rute1, rute2];
```

Vi antar at alle bussruter har nøyaktig de samme stoppestedene i begge retninger, slik at vi i ruteheftet bare angir stoppestedene den ene veien.

I oppgavene nedenfor skal du kun bruke den delen av ML som er pensum, og alle funksjoner skal være rent funksjonelle. Lag gjerne hjelpefunksjoner der det er naturlig.

4a Funksjonen steder

Skriv en funksjon

```
fun steder(nr:int, rh:rutehefte):string list = ...;
```

som gir en liste med alle stoppestedene til bussen med rutenummer nr i ruteheftet rh.

(Fortsettes på side 7.)

4b Funksjonen busser

Skriv en funksjon

```
fun busser(st:string, rh:rutehefte):int list = ...;
```

som finner nummeret til alle bussene i rh som stopper ved stoppestedet angitt av st.

4c Ny bussrute

Skriv en funksjon

```
fun leggitil(nr:int, ss:string list,
            rh:rutehefte):rutehefte = ...;
```

som tar en bussrute med rutenummer nr og stoppesteder ss og legger denne til i ruteheftet rh. Hvis det allerede finnes en bussrute med dette rutenummeret i rh, skal den gamle ruten fjernes fra ruteheftet.

4d Nedlegging av stoppested

Skriv en funksjon

```
fun fjern(st:string, rh:rutehefte):rutehefte = ...;
```

som legger ned stoppestedet angitt ved st (dvs. fjerner dette stoppestedet fra alle rutene i rh).

4e Bytte mellom busser

Hvis to busser har et felles stoppested, antar vi at det er mulig å bytte mellom dem på dette stedet.

Skriv en funksjon

```
fun maksEttBytte(A:string, B:string,
                rh:rutehefte):bool = ...;
```

som sjekker om det er mulig å reise fra A til B med maksimalt ett bytte.

(Fortsettes på side 8.)

4f Implementasjon ved funksjonsrom

Vi skal nå istedenfor liste-strukturen over representere bussruter og rutehefter ved hjelp av funksjonsrom, på følgende måte:

```
type rutehefte = int -> string list;
```

der et rutehefte `rh` er funksjonen som for et gitt rutenummer gir listen med stoppesteder som funksjonsverdi.

1. Skriv funksjonene `steder`, `leggtil` og `fjern` med denne nye representasjonen av rutehefter.
2. Hvorfor er det ikke mulig å lage funksjonen `busser` med denne nye representasjonen? Forklar kort.

Oppgave 5 Bussruter i Prolog (vekt 10%)

I Prolog kan bussruter representeres ved fakta av typen `stopper(bussrute, stoppested)`.

Eksempler på slike fakta er:

```
stopper(23, lysaker).
stopper(23, nygård).
stopper(23, teisen).
stopper(23, ryen).

stopper(21, akerbrygge).
stopper(21, helsfyr).
stopper(21, teisen).
stopper(21, tveita).

stopper(37, helsfyr).
stopper(37, stortorget).
stopper(37, nygård).
stopper(37, kjelsås).
```

Du skal i det følgende holde deg til de deler av Prolog som er pensum (det er heller ikke nødvendig å bruke liste-konstruksjonen).

5a Oppslag

Skriv et spørsmål som finner alle rutene som stopper ved "helsfyr".

(Fortsettes på side 9.)

5b Knutepunkter

Lag en regel som angir om et stopp er et knutepunkt (dvs. at det stopper minst to busser der).

Skriv et spørsmål som gir alle knutepunkter som svar. Gitt fakta som over, hva blir svaret på dette spørsmålet og hvordan er det beregnet? Svar kort.

5c Reiserute

Gi regler og spørsmål som gitt to stoppesteder angir om det er mulig å reise mellom dem med maksimalt ett bytte av buss. Som svar på spørsmålet skal man få nummeret på de aktuelle bussene.

(Slutt oppgavetekst.)